
An efficient algorithm for Nonlinear Active Noise Control of Impulsive Noise

Z. C. He^a, H. H. Ye^{a*}, Eric Li^{b*}

^aState Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University,
Changsha, 410082 P. R. China

^bSchool of Science, Engineering & Design, Teesside University, Middlesbrough, UK

Abstract

Nonlinear active noise control (NANC) systems employing Volterra filter suffer from the stability issues in the presence of impulsive noise. To solve this problem, we combine the second-order Volterra (SOV) filter and maximum correntropy criterion (MCC) in this paper. The Volterra filter-x maximum correntropy criterion (VFxMCC) algorithm and Volterra filter-x recursive maximum correntropy (VFxRMC) algorithm are applied to reduce the impulsive noise of NANC. We find that VFxMCC algorithm has a low computational complexity and VFxRMC algorithm converges fast. In order to extract their advantages, we further propose a hybrid algorithm based on the VFxMCC and VFxRMC algorithms. In addition, the normalize step-size version of VFxMCC (VFxnMCC) algorithm is developed to improve the robustness and performance. Meanwhile, we adaptively adjust the kernel size of MCC online based on the sample variance of reference signal to improve the performance of the proposed algorithms. Simulation results in the context of nonlinear active impulsive noise control demonstrate that the proposed algorithms achieve much better performance than the existing algorithms in various noise environments.

Keywords: nonlinear active noise control; impulsive noise; maximum correntropy criterion; adaptive Volterra filter;

* Corresponding author.

E-mail address: 470894202@qq.com (HH Ye)

ericsg2012@gmail.com (Eric Li)

1. Introduction

Active noise control (ANC) is based on the superposition principle of the primary noise source and secondary source with their acoustic outputs being the same amplitude but having opposite phase [1-3]. The ANC system based on the filter-x least mean square (FxLMS) algorithm provides a satisfactory level of performance in linear system with Gaussian noise [4, 5]. However, the primary paths always tend to be non-linear [6, 7] and the main noise is non-Gaussian noise in practice. As a common non-Gaussian noise, the impulsive noise is a signal that contains unwanted, almost instantaneous and abrupt mutations. In order to improve the robustness of non-linear ANC (NANC) in the case where the reference noise is impulsive noise, many studies have been done in the past [8-13]. In 2004, Das et al. developed a new filtered-s LMS (FsLMS) algorithm using a single-layer functional linked artificial neural network (FLANN) as a controller for NANC [12]. Tan et al. proposed an adaptive Volterra FxLMS (VFxLMS) algorithm [11] based on a multichannel structure for feedforward nonlinear active noise control. The Volterra filter is composed of multiple orders, and the input signal is expanded by each order using different mathematical methods, resulting in a high computational complexity of the Volterra-based algorithms. Due to the bottleneck of calculation, second-order Volterra (SOV) filters are often used in practical applications. In addition, several computationally efficient Volterra filters were developed by using the different structures for the NANC systems [14, 15]. In 2012, George et al. developed a robust filtered-s LMS (RFsLMS) NANC algorithm [10]. In addition, a Volterra filter-x with continuous least mean l_p -norm using logarithmic cost (VFxlogCLMP) algorithm was proposed by Lu et al. to improve the stability in the impulsive noise environment [16].

Recently, the information entropy (IE) method was applied to the active noise control system for the impulsive noise and good results were achieved [17-20]. Correntropy of IE is a nonlinear and local similarity measure between two arbitrary scalar random variables. The maximum correntropy criterion (MCC), aiming at maximizing the similarity of primary noise and secondary noise, can effectively improve the robustness of ANC. Nikhil et al. developed a correntropy based algorithm, which improves the robustness of NANC system

for non-Gaussian reference noise signals [19]. In 2017, Lu et al. proposed a filtered-x recursive maximum correntropy (FxRMC) algorithm based on MCC to reduce the effect of impulsive noise[18].

In this paper, we proposed nonlinear adaptive algorithms based on second-order Volterra expansion model and MCC, which is called the VFxMCC and VFxRMC algorithms. The proposed algorithms have great potential used in a wide range of applications involving impulsive noise in NANC. Similar to the LMS-based algorithms [21, 22], the VFxMCC algorithm uses the gradient descent theory to update the weight vector. Moreover, the weight coefficient updating mode of VFxRMC algorithm is parallel to the recursive least square (RLS) based algorithms [23-25]. In order to further improve algorithm performance, we use the VFxRMC algorithm to update the 1st-order SOV filter coefficient and the 2nd-order SOV filter coefficient is updated by VFxMCC algorithm, which is called the hybrid (HVFx-RMC-MCC) algorithm. The proposed hybrid algorithm has a low computational complexity and fast convergence. In particular, our main contributions are depicted as the following points:

1. The proposed NANC algorithms can work effectively with impulsive non-Gaussian noise in non-linear system, while lots of existing algorithms are only applicable to impulsive noise linear systems or Gaussian noise non-linear systems.
2. The normalized step-size VFxMCC (VFxnMCC) is developed to improve the robustness of algorithm by adjusting the step-size with the estimate of input signal power. Moreover, we independently choose the different step-size to update each Volterra filter weight vector.
3. As the MCC kernel size has a great effect on the performance of MCC-based algorithms [18], we estimate the sample variance of reference noise to adjust the MCC kernel size of the proposed algorithms motivated by the thumb rule [26] of MCC kernel size selection. In addition, we set a threshold for MCC kernel size to enhance the robustness.
4. The proposed hybrid algorithm has a lower computational burden than the VFxRMC algorithm, and converges faster than the VFxnMCC algorithm, which is one of the main innovations of this article.

The rest of the paper is organized as follows: Section 2 briefly introduces preliminary work of Volterra filter and MCC. The proposed algorithms in this paper are presented in Section 3. Section 4 describes the analysis of proposed algorithms, including the convergence and computational complexity. The simulations studies are carried out to evaluate the performance of the three proposed algorithms in Section 5. Finally, Section 6 presents the discussion and conclusion of this work.

2. The theory of Volterra filter and MCC

2.1 Volterra filter of NANC

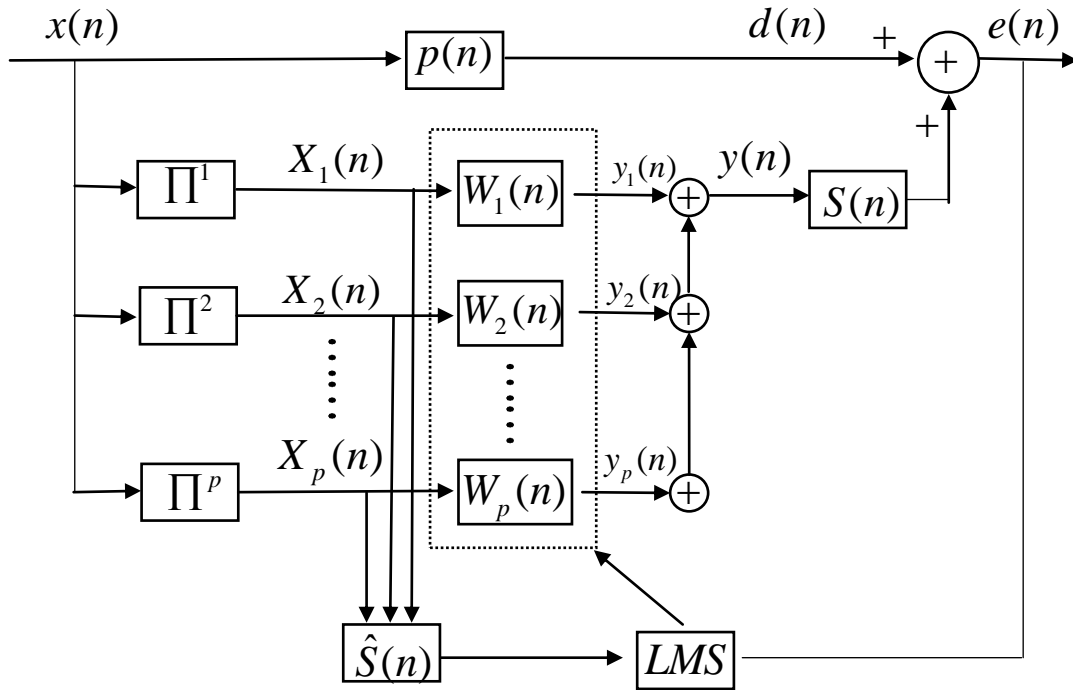


Fig.1 The block diagram using Volterra expansion for NANC

It is well-known that Volterra filters have an excellent performance in noise reduction of NANC. The block diagram using Volterra filter for NANC is shown in Fig.1. In addition, $P(z)$ denotes the transfer function of the primary path from the noise source to the cancellation point; Similarly, the transfer function of the secondary path (path from the controller output to the cancellation point) is represented by $S(z)$, and its estimate $\hat{S}(z)$ is assumed to be the same. $x(n)$ is the reference noise measured by the reference microphone, and the residual noise $e(n)$ is picked up by the error microphone. Furthermore, the reference

noise $x(n)$ is expanded by Volterra filter to get the total input signal $X(n)$ of ANCA. $X_p(n)$ denotes the P^{th} order filter expanded input signal, where P represents the order of Volterra filter [11]. For secondary-order Volterra filter, P takes a value of 1 or 2. The input signals can be written as

$$X_1(n) = \sum_{i=0}^{L-1} x(n-i) = [x(n), x(n-1), \dots, x(n-L+1)], \quad (1)$$

$$\begin{aligned} X_2(n) &= \sum_{i=0}^{L-1} \sum_{j=i}^{L-1} x(n-i)x(n-j) \\ &= [x^2(n), x(n)x(n-1), \dots, x(n)x(n-L+1), x^2(n-1), \dots, x^2(n-L+1)], \end{aligned} \quad (2)$$

$$X(n) = [X_1(n), X_2(n)], \quad (3)$$

where n represents the time index and L is the length of the reference noise $x(n)$. Obviously, the dimensions of $X_1(n)$ and $X_2(n)$ are $1 \times L$ and $1 \times M$. It is noted that $M = (L+1)L/2$. $W_p(n)$ is the P^{th} order filter adaptive weight vector, and it can be expressed as

$$W_1(n) = \sum_{i=0}^{L-1} w_1(n-i) = [w_1(n), w_1(n-1), \dots, w_1(n-L+1)], \quad (4)$$

$$\begin{aligned} W_2(n) &= \sum_{i=0}^{L-1} \sum_{j=i}^{L-1} w_2(n-i, n-j) \\ &= [w_2(n, n), w_2(n, n-1), \dots, w_2(n-L+1, n-L+1)], \end{aligned} \quad (5)$$

$$W(n) = [W_1(n), W_2(n)]. \quad (6)$$

$W(n)$ denotes the total SOV filter adaptive weight vector. The dimension of $W_1(n)$ is $1 \times L$, and the dimension of $W_2(n)$ is $1 \times M$. By using the second-order Volterra expansion, the output of adaptive nonlinear controller $y(n)$ can be written as

$$\begin{aligned} y(n) &= W(n)X^T(n) \\ &= \sum_{i=0}^{L-1} x(n-i)w_1(n-i) + \sum_{i=0}^{L-1} \sum_{j=i}^{L-1} x(n-i)x(n-j)w_2(n-i, n-j), \end{aligned} \quad (7)$$

In addition, the residual noise is given by

$$e(n) = d(n) - s(n) * y(n); \quad (8)$$

$d(n)$ denotes the desired signal in cancellation point. In traditional NANC systems, the most common weight updating strategy is based on the gradient descent method that minimizes mean square error (MSE). Therefore, the weight updating rule of VFxLMS [11] algorithm is given by

$$W(n+1) = W(n) + 2\mu X'(n)e(n) \quad (9)$$

where the signal vector $X'(n)$ contains all filtered signal elements corresponding to all input elements in $X(n)$. It can be obtained as

$$X'(n) = X(n) * \hat{s}(n), \quad (10)$$

where $*$ indicates convolution.

2.2 MCC

Correntropy is a nonlinear and local similarity measure between two arbitrary scalar random variables α and β . Then, it can be defined by [17].

$$V(\alpha, \beta) = E[K_\sigma(\alpha - \beta)] = \int K_\sigma(\alpha, \beta) f_{\alpha, \beta}(\alpha, \beta) d\alpha d\beta, \quad (11)$$

where $K_\sigma(\alpha, \beta)$ is a Gaussian kernel. $f_{\alpha, \beta}(\alpha, \beta)$ is the joint probability density function of α and β . The commonly used Gaussian kernel $K_\sigma(\alpha, \beta)$ [27] is defined as

$$K_\sigma(\alpha - \beta) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(\alpha - \beta)^2}{2\sigma^2}\right), \quad (12)$$

where σ denotes the kernel size. In practice, we cannot obtain the joint probability density function. Therefore, a finite number of data $\{((\alpha_i, \beta_i))\}_{i=1}^N$ are available, leading to the sample estimator of correntropy [28]

$$\hat{V}(\alpha, \beta) = \frac{1}{N} \sum_{i=1}^N K_\sigma(\alpha_i - \beta_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(\alpha - \beta)^2}{2\sigma^2}\right). \quad (13)$$

MCC expects a maximum value of correntropy and describes the correlation between α and β . As $\alpha = \beta$, MCC obtains the maximum value. Therefore, a larger value of

correntropy indicates a higher similarity between the two random signals [17] .

3. The idea of the proposed algorithms

3.1 A brief of the VFxmMCC algorithm

When it comes to active noise control, we always expect a small residual noise. As we can see, the residual noise $e(n)$ is equal to the difference between the desired signal $d(n)$ and output signal $y(n)$. Based on the MCC, the cost function of the VFxmMCC algorithm can be defined as

$$\zeta_n = \hat{V}(D, Y) = \sum_{i=1}^N \exp\left(-\frac{(d(i) - y(i))^2}{2\sigma^2}\right) = \sum_{i=1}^N \exp\left(-\frac{e^2(i)}{2\sigma^2}\right) \approx \exp\left(-\frac{e^2(n)}{2\sigma^2}\right) \quad (14)$$

The weight updating rule using gradient ascent approach can be written as

$$W(n+1) = W(n) + \mu \nabla \zeta_n; \quad (15)$$

In addition, the gradient of the cost function is given by

$$\nabla \zeta_n = \frac{\partial \zeta_n}{\partial W} \approx e(n) X'(n) \exp\left(-\frac{e^2(n)}{2\sigma^2}\right); \quad (16)$$

We defined $Q(n) = \exp\left(-\frac{e^2(n)}{2\sigma^2}\right)$ and get the weight updating rule of the VFxmMCC algorithm as

$$W(n+1) = W(n) + \mu X'(n) e(n) Q(n). \quad (17)$$

μ is the step-size of VFxmMCC algorithm which can seriously affect the weight updating rule [29]. What is more, the filter weight vector $W(n)$ of SOV filter is combined by two layers of weight vectors $W_p(n)$. Therefore, we can choose different μ for the updating rule of each filter weight vector independently. Finally, the weight updating of proposed algorithm VFxmMCC can be rewritten as

$$W_p(n+1) = W_p(n) + \mu_p e(n) X'(n) Q(n); \quad (18)$$

The value of the step-size μ_p is related to the input signal power of Volterra filter. To improve the robustness, the normalized step-size version of VFxmMCC (VFxmMCC) is

developed, which is to take the time-varying μ_p . The step-size is no longer fixed but changes with the on-line power estimation of the input signal [30]. In addition, we use the sliding rectangular window method to estimate the input signal power. In addition, the length L_p of the window is equal to the input signal length. This means $L_1 = L$ and $L_2 = M$.

The power estimation can be defined as

$$P_p(n) = \sum_{m=0}^{L_p-1} x^2(n-m) = [X'_p(n)X_p'^T(n)]; \quad (19)$$

Then, the time-varying step-size can be written as

$$\mu_p(n) = \frac{\mu_p}{X'_p(n)X_p'^T(n)}; \quad (20)$$

Finally, the weight updating of VFxnMCC algorithm can be expressed as

$$W_p(n+1) = W_p(n) + \mu_p(n)e(n)X'_p(n)Q(n); \quad (21)$$

3.2 A brief of the VFxRMC algorithm

We take the similar cost function in VFxnMCC algorithm to combine Volterra filter and MCC with RLS-based updating mode. Therefore, the cost function of VFxRMC algorithm is defined as

$$\zeta_n = \sum_{i=1}^n \lambda^{n-i} \exp\left(-\frac{e^2(i)}{2\sigma^2}\right), \quad (22)$$

where $\lambda(0 < \lambda < 1)$ is the forgetting factor. The gradient of the cost function can be given by

$$\nabla \zeta_n = \frac{\partial \zeta_n}{\partial W} = \sum_{i=1}^n \lambda^{n-i} \frac{e(i)X'(i)}{2\sigma^2} \exp\left(-\frac{e^2(i)}{2\sigma^2}\right) = \sum_{i=1}^n \lambda^{n-i} \frac{e(i)X'(i)Q(i)}{2\sigma^2}; \quad (23)$$

Previous researchers [18] have given the RLS-based updating mode of MCC applied to active noise control. Then, we can summarize the VFxRMC algorithm as follows

$$G_p(n) = \frac{Q(n)X'_p(n)\mathbf{P}_p(n-1)}{Q(n)X'_p(n)\mathbf{P}_p(n-1)X_p'^T(n) + \lambda}, \quad (24)$$

$$\mathbf{P}_p(n) = \lambda^{-1}\mathbf{P}_p(n-1) - \lambda^{-1}G_p(n)X_p'^T(n)\mathbf{P}_p(n-1), \quad (25)$$

$$W_p(n+1) = W_p(n) + G_p(n)e(n), \quad (26)$$

where $G_p(n)$ is the gain vector of P^{th} order filter. $\mathbf{P}_p(n)$ is the inverse of the correlation matrix. What is more, we can get that the dimension of $\mathbf{P}_1(n)$ is $L \times L$ and the dimension of $\mathbf{P}_2(n)$ is $M \times M$. And, the dimension of $G_p(n)$ is the same as the filter expanded signal $X_p(n)$. Therefore, we can conclude that their dimensions depend on the length of the input signal. The length of input signal becomes longer after Volterra filter expansion leading to a greater computational burden of VFxRMC algorithm. Nevertheless, the VFxRMC has a very fast convergence rate, which is a significant characteristic of ANC in practice.

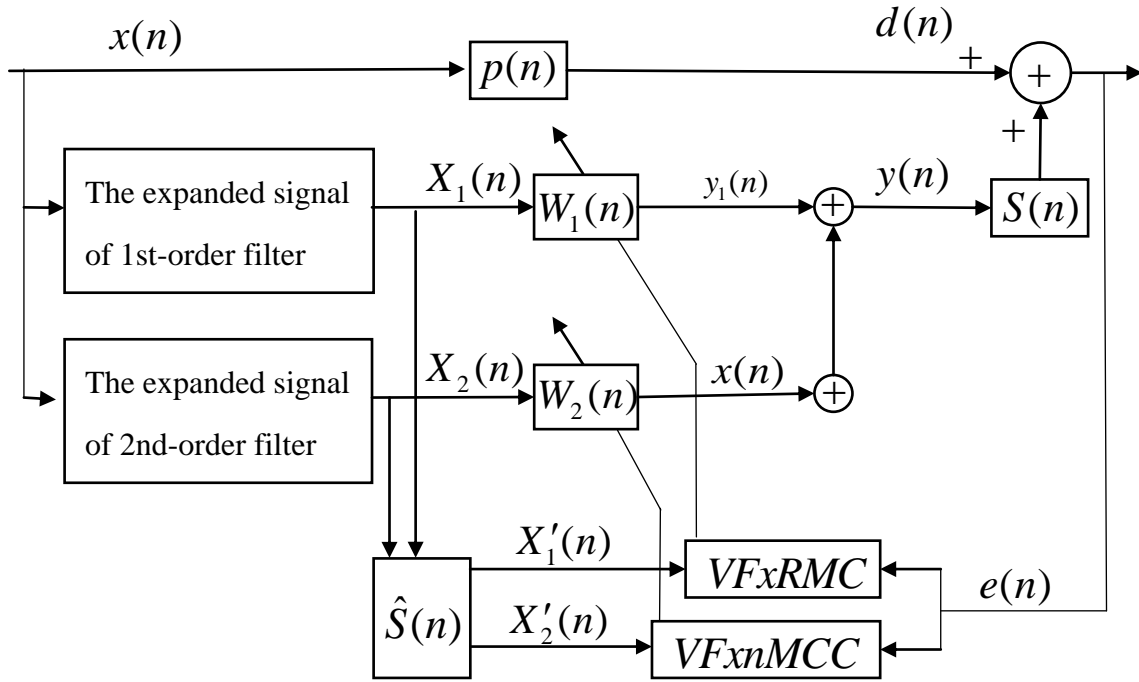


Fig.2 The block diagram of HVFx-MCC-RMC algorithm

3.3 The proposed hybrid algorithm

The proposed hybrid algorithm is shown in Fig.2. We use the VFxRMC algorithm to update the 1st-order filter coefficient $W_1(n)$, while the 2nd-order filter coefficient $W_2(n)$ is updated by VFxnMCC algorithm. Therefore, the weight updating of hybrid HVFx-MCC-RMC algorithm can be obtained by

$$W_1(n+1) = W_1(n) + \mu_1(n)e(n)X_1'(n)Q(n), \quad (27)$$

$$W_2(n+1) = W_2(n) + G_2(n)e(n);$$

In the secondary-order Volterra filter, the length of the 1st-order input signal $X_1(n)$ is L , and the 2nd-order is M . Due to the short length of input signal $X_1(n)$, the computational complexity gain caused by the VFxRMC algorithm in 1st-order filter is little and acceptable. In this way, the hybrid algorithm has a lower computational burden than the VFxRMC algorithm, and converges faster than the VFxnMCC algorithm. We will carefully analyze and compare the computational complexity of the proposed algorithms. In addition, the simulation test will show the performance of the proposed algorithms.

3.4 Adaptive Gaussian kernel size of MCC-based algorithms

As shown in Fig. 3, the cost function of MCC-based algorithms changes with diverse kernel size. MCC belongs to a kind of probability density estimation method, and the Gaussian kernel size can affect the estimation accuracy [17]. The probability density of the reference signals in ANC changes randomly, resulting in the need of a time-varying kernel size.

The kernel size is a free parameter that must be chosen by the user using the concepts of density estimation [17]. A thumb rule [26] was proposed by Silveman to estimate kernel size which is expressed as follows:

$$\sigma = \left(\frac{4}{3n}\right)^{1/5} \hat{\sigma}; \quad (28)$$

$$\hat{\sigma} = \frac{1}{L-1} \left(\sum_{i=1}^L x_i^2 - L\bar{x}^2 \right), \quad (29)$$

where $\hat{\sigma}$ denotes the signal sample variance. Motivated by the thumb rule, we recursively update the kernel size based on the sample variance, which can be given by

$$\sigma^2(n+1) = \gamma\sigma^2(n) + (1-\gamma)\hat{\sigma}, \quad (30)$$

where γ ($0 < \gamma < 1$) is close to 1, and $\hat{\sigma}$ is the sample variance of the reference signal $x(n)$ of ANC. Obviously, the size of $\sigma^2(n)$ is proportional to the value of reference sample

variance. As the heavy impulsive reference may cause a large size of $\sigma^2(n)$, it will weaken the stability of algorithm. Therefore, it is necessary to set a threshold Ψ for $\sigma^2(n)$, which can be defined as

$$\sigma^2(n) \leq \Psi. \quad (31)$$

where Ψ is found based on the actual situation.

Finally, the $Q(n)$ of weight vector in the adaptive VFxnMCC (A-VFxnMCC), the adaptive VFxRMC (A-VFxRMC) and the adaptive HVFx-MCC-RMC (A-HVFx-MCC-RMC) algorithms can be rewritten as

$$Q(n) = \exp\left(-\frac{e^2(n)}{2\sigma^2(n)}\right); \quad (32)$$

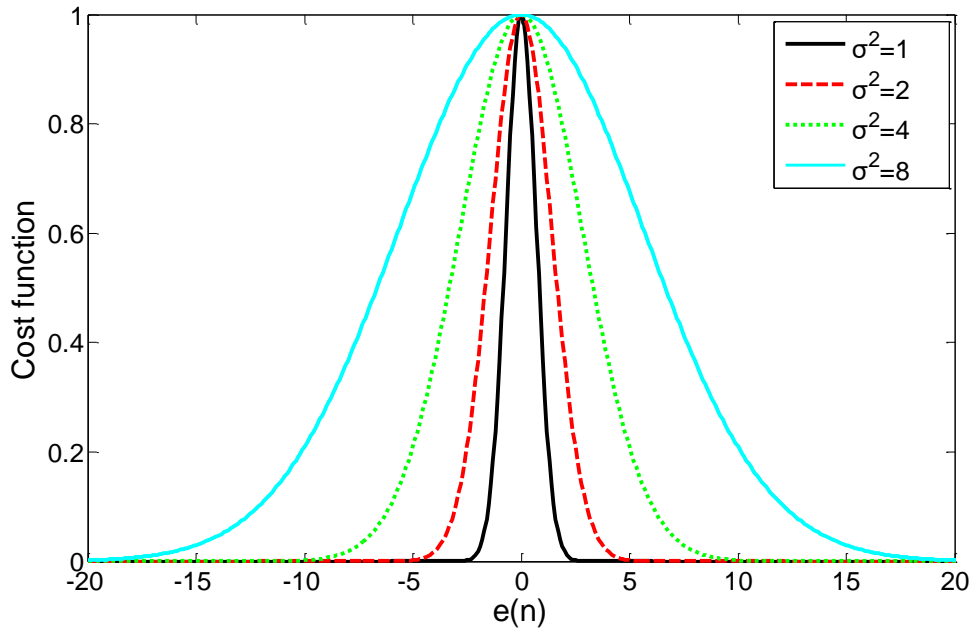


Fig.3. The cost functions with different σ^2 .

4. Analysis of the proposed algorithms

4.1 Mean convergence of A-VFxnMCC algorithm

First, we consider W_{p_0} as the optimal weight vector of the P^{th} adaptive filter and the deviation of the weight vector from W_{p_0} is

$$\hat{W}_p(n) = W_{p0} - W_p(n). \quad (33)$$

We can combine Eqs. (21) and (33) to get

$$\hat{W}_p(n+1) = \hat{W}_p(n) - \frac{\mu_p}{X_p'(n)X_p'^T(n)} e(n)X_p'(n) \exp\left(-\frac{e^2(n)}{2\sigma^2(n)}\right); \quad (34)$$

Like [10], we assume $\Phi(n)$ as the expectation of the squared Euclidean norm i.e.

$\Phi(n) = E\left(\left\|\hat{W}_p(n)\right\|^2\right)$. Therefore, we can get

$$\begin{aligned} \Phi(n+1) &= \Phi(n) + \mu_p^2 E\left\{\frac{e(n)X_p'(n)}{X_p'(n)X_p'^T(n)} \exp\left(-\frac{e^2(n)}{2\sigma^2(n)}\right)\right\}^2 \\ &\quad - 2\mu_p E\left\{\frac{e(n)X_p'^T(n)\hat{W}_p(n)}{X_p'(n)X_p'^T(n)} \exp\left(-\frac{e^2(n)}{2\sigma^2(n)}\right)\right\} \end{aligned} \quad (35)$$

For the purpose of the stability and convergence, $\Phi(n+1)$ should be less than $\Phi(n)$.

Besides, we consider $e(n) = X_p'(n)\hat{W}_p(n)^T$. Hence, the bounds for μ_p are obtained as

$$0 < \mu_p < \frac{2}{E\left\{\exp\left(-\frac{e^2(n)}{2\sigma^2(n)}\right)\right\}} \quad (36)$$

4.2 Mean convergence of A-VFxRMC algorithm

We can combine Eqs. (26) and (33) to get

$$\hat{W}_p(n+1) = \hat{W}_p(n) - \frac{Q(n)X_p'(n)P_p(n-1)}{Q(n)X_p'(n)P_p(n-1)X_p'^T(n) + \lambda} e(n); \quad (37)$$

Taking expectation of both sides of absolute value in Eq. (37), it can be rewritten as

$$E\left\{\left\|\hat{W}_p(n+1)\right\|\right\} = E\left\{\left\|\hat{W}_p(n)\right\|\right\} E\left\{\left|1 - \frac{Q(n)X_p'(n)P_p(n-1)X_p'^T(n)}{Q(n)X_p'(n)P_p(n-1)X_p'^T(n) + \lambda}\right|\right\}, \quad (38)$$

assuming $e(n) = X_p'(n)\hat{W}_p(n)^T$. To ensure the stability and convergence, $E\left\{\left\|\hat{W}_p(n+1)\right\|\right\}$ should be less than $E\left\{\left\|\hat{W}_p(n)\right\|\right\}$. Therefore, the weight vector in the A-VFxRMC algorithm converges if and only if

$$0 < E \left\{ \left| \frac{Q(n)X'_p(n)\mathbf{P}_p(n-1)X'^T_p(n)}{Q(n)X'_p(n)\mathbf{P}_p(n-1)X'^T_p(n) + \lambda} \right| \right\} < 1 \quad (39)$$

Hence, the mean weight vector of the A-VF_xRMC algorithm is convergent when the input signal is persistently exciting [31].

4.3 Computational complexity analysis

The computational complexity of algorithms can seriously affect the performance of noise reduction. In practical applications, the ANC system needs to meet the real-time and fast computation performance. Large computational complexity not only increases the requirements on the hardware system but also affects the system convergence rate.

The summary of the A-VF_xnMCC and A-VF_xRMC algorithms is given in Tables 1 and 2. The computational complexity for the adaptive filters is summarized in Table 3.

Table 1

Summary of A-VF_xnMCC algorithm

Eq.'s	Operations
(1).	$X'(n) = s(n) * X(n);$
(2).	$y(n) = X(n)W^T(n);$
(3).	$e(n) = d(n) - y(n);$
(4).	$\bar{x} = \frac{1}{L} \sum_{i=1}^L x_i;$
(5).	$\hat{\sigma} = \frac{1}{L-1} (\sum_{i=1}^L x_i^2 - L\bar{x}^2);$
(6).	$\sigma^2(n+1) = \gamma\sigma^2(n) + (1-\gamma)\hat{\sigma};$
(7).	$W_p(n+1) = W_p(n) + \mu_p(n)e(n)X'_p(n)\exp(-\frac{e^2(n)}{2\sigma^2(n)});$
(8).	$\mu_p(n) = \frac{\mu_p(n)}{X'_p(n)X'^T_p(n)};$

Table 2

Summary of A-VF_xRMC algorithm

Eq.'s	Operations
-------	------------

-
- (1). $X'(n) = s(n) * X(n);$
 - (2). $y(n) = X'(n)W^T(n);$
 - (3). $e(n) = d(n) - y(n);$
 - (4). $\bar{x} = \frac{1}{L} \sum_{i=1}^L x_i;$
 - (5). $\hat{\sigma} = \frac{1}{L-1} (\sum_{i=1}^L x_i^2 - L\bar{x}^2);$
 - (6). $\sigma^2(n+1) = \gamma\sigma^2(n) + (1-\gamma)\hat{\sigma};$
 - (7). $Q(n) = \exp(-\frac{e^2(n)}{2\sigma^2(n)});$
 - (8). $G_p(n) = \frac{Q(n)X'_p(n)\mathbf{P}_p(n-1)}{Q(n)X'_p(n)\mathbf{P}_p(n-1)X_p'^T(n) + \lambda};$
 - (9). $\mathbf{P}_p(n) = \lambda^{-1}\mathbf{P}_p(n-1) - \lambda^{-1}G_p(n)X_p'^T(n)\mathbf{P}_p(n-1);$
 - (10). $W_p(n+1) = W_p(n) + G_p(n)e(n);$
-

As we can see in the Table 3, the computational complexity of LMS-based (VFxLMS, A-VFxnMCC) algorithms is similar and low, while the A-VFxRMC has a heavy computation burden. What is more, the computational complexity of A-HVFx-RMC-MCC algorithm is much lower than that of A-VFxRMC algorithm, since L is much shorter than M .

Table 3

Computational complexity of the algorithms for ANC.

Algorithms	Multiplications	Additions/Subtractions	Divisions	Exponential operations
VFxLMS	$3(M+L)+2$	$3(M+L)-2$	2	-
A-VFxnMCC	$3(M+L)+6$	$3M+5L+2$	5	1
A-VFxRMC	$4(M^2+L^2)+5(M+L)+2$	$3(M^2+L^2)+6M+8L+6$	8	1

A-HVFX-RMC -MCC	$4L^2 + 5L + 3M + 5$	$3L^2 + 5M + 3$	6	1
--------------------	----------------------	-----------------	---	---

Note:

L denotes the length of $X_1(n)$, and, M is the length of $X_2(n)$

5. Simulation results

We consider the impulsive noise with symmetric α -stable (SaS) distribution. The SaS process has no closed probability density function expression, and can be described using the characteristic function [32]

$$\varphi(n) = \exp(-|t|^\alpha). \quad (40)$$

$\alpha (0 < \alpha < 2)$ is the characteristic exponent describing the impulsive signal level. A small value of α indicates a peaky and heavy tailed distribution with high outliers.

A host of scholars have studied the nonlinear active noise control in the presence of impulsive noise such as the famous RFslms algorithm [10]. And the weight of this controller is given by

$$W(n+1) = W(n) + \mu(n) \frac{e(n)}{e^2(n) + 2\sigma^2} X'(n); \quad (41)$$

Next, we carry out the simulation to verify the effectiveness of the proposed algorithms in comparison with the VFxLMS, VFxlogCLMP [16], VFxlogLMS [33] and RFsLMS algorithms. The averaged noise reduction (ANR) is used to compare the performance of ANC, which is defined as

$$ANR(n) = 20 \log\left(\frac{A_e(n)}{A_d(n)}\right), \quad (42)$$

where $A_e(n) = \xi A_e(n-1) + (1-\xi)|e(n)|$, $A_d(n) = \xi A_d(n-1) + (1-\xi)|e(n)|$. Note that $\xi = 0.99$ is the forgetting factor.

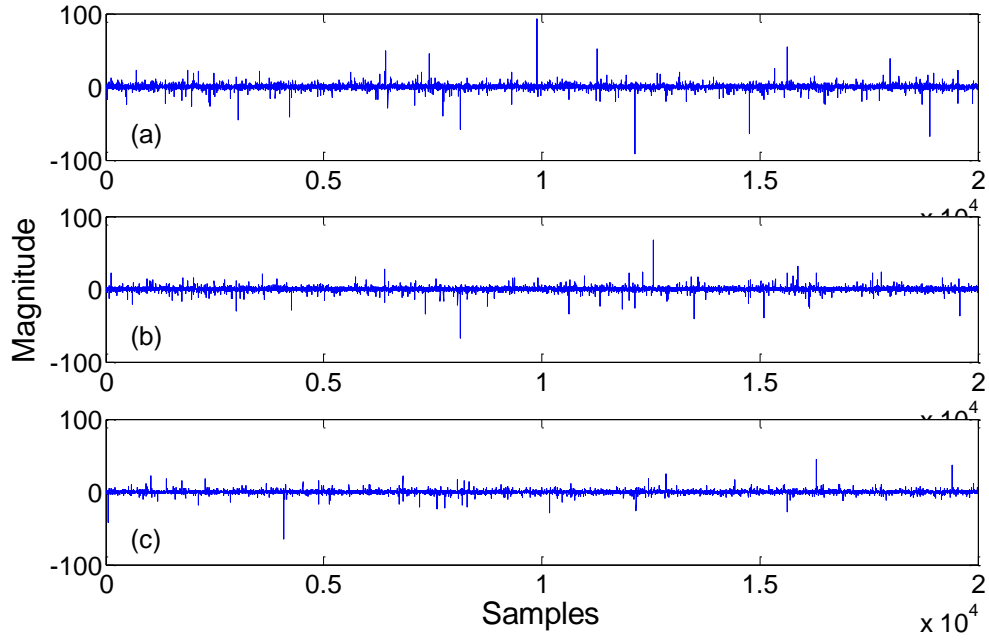
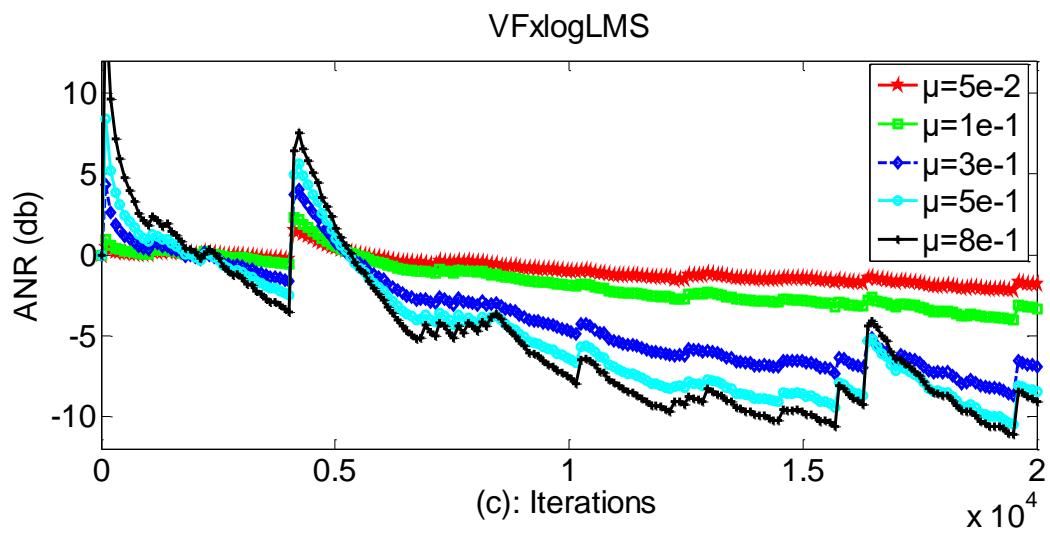
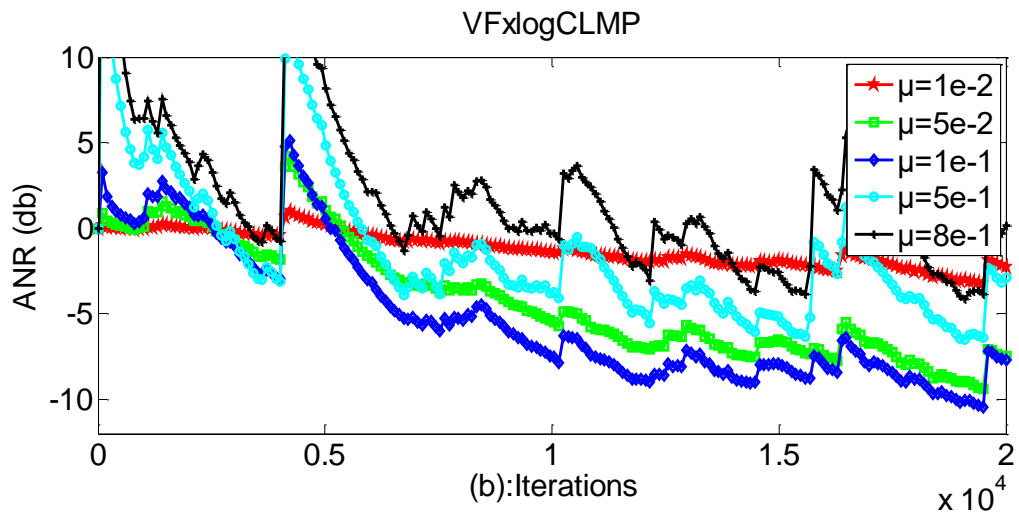
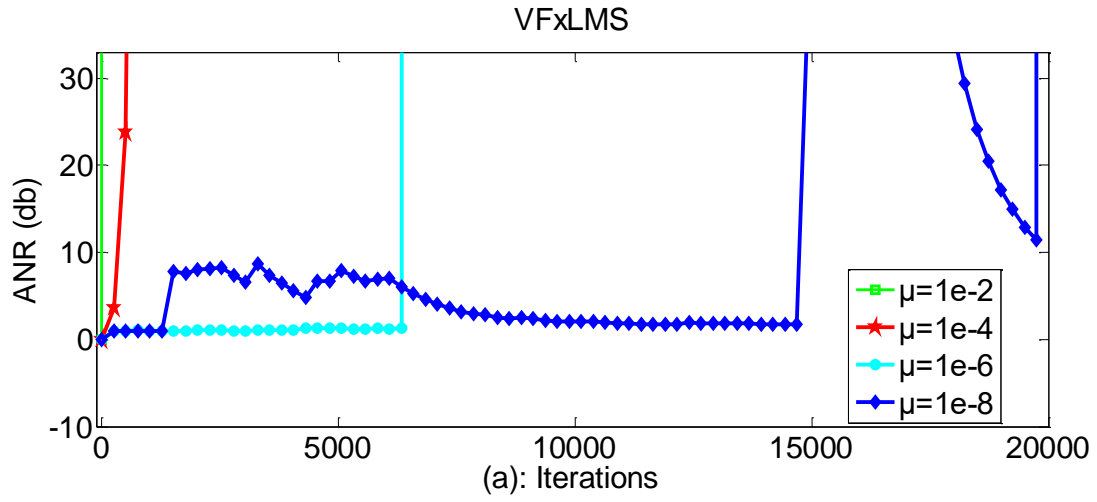
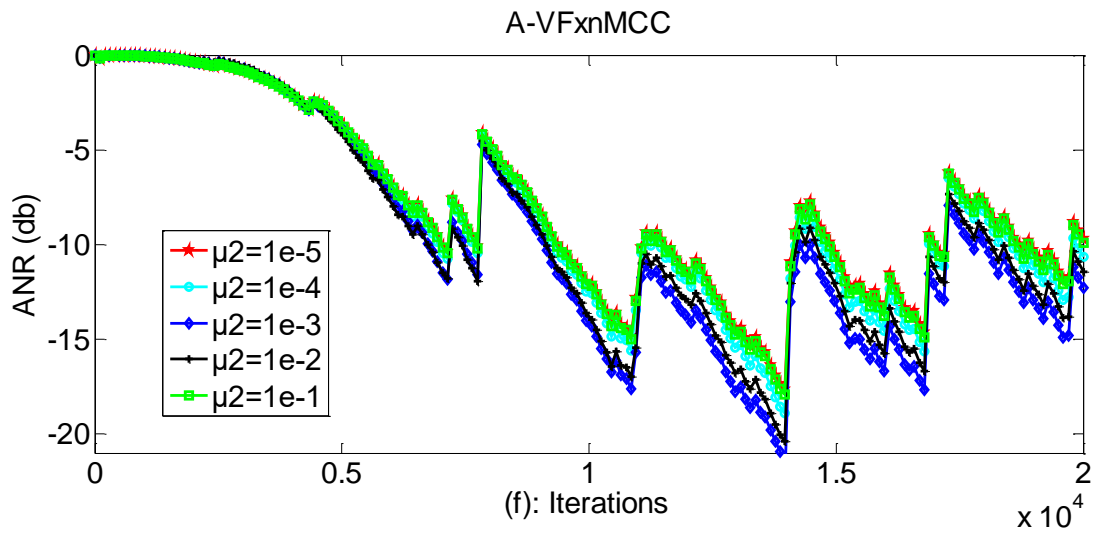
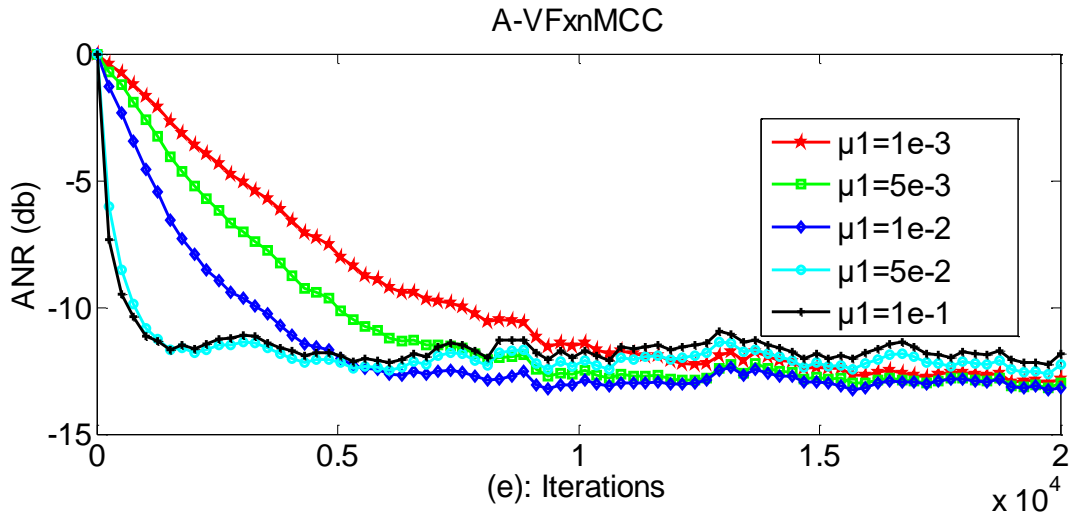
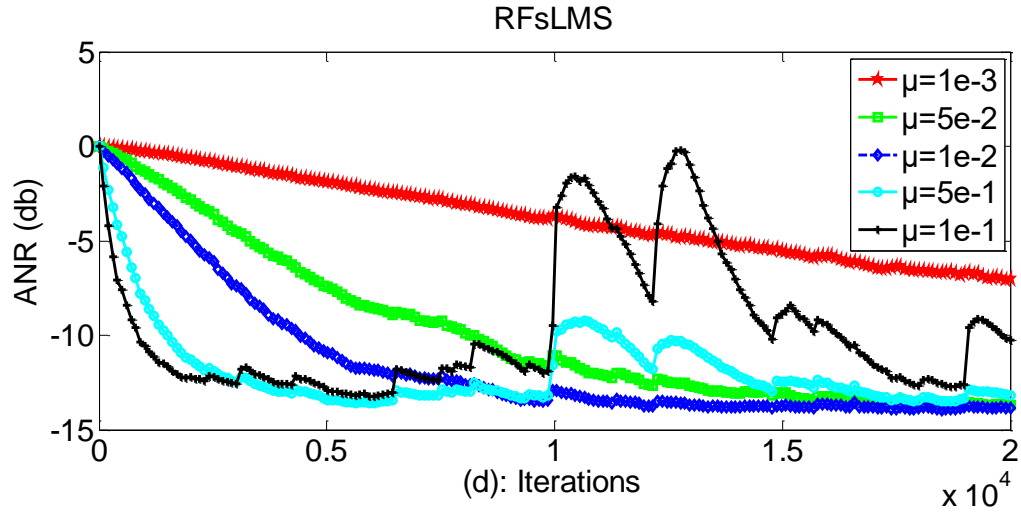


Fig.4 Reference noise of experiment 1. case 1 (a) $\alpha = 1.7$, case 2 (b) $\alpha = 1.8$, case 3(c) $\alpha = 2.0$.

5.1 Experiment 1: Non-minimum phase secondary path

In this experiment, we use the reference noise $x(n)$ having a $S\alpha S$ distribution with $\alpha=1.7$ and $\alpha=1.8$. The primary path $Q(z)=z^{-3}-0.3z^{-4}+0.2z^{-5}$ is the modified FIR filter. We consider the secondary transfer function to be non-minimum phase as $S(z)=z^{-2}+1.5z^{-3}-z^{-4}$. Fig. 4 shows the original reference noise of experiment 1. The detailed simulation results for the effects of step-size μ on the relevant algorithms are presented in Fig. 5(a-f). The selection of forgetting factor λ on A-VFRMC algorithm is shown in Fig. 5(g). Similarly, the effects of threshold Ψ of the proposed hybrid algorithm are displayed in Fig. 5(h). As we can see, the threshold value can affect the convergence rate and the stability of the proposed algorithm. The low value of threshold reduces the convergence rate, while the high value degenerates the stability. However, the precise choice of Ψ is based on the actual condition. In this paper, the proposed hybrid algorithm performs best in Fig. 5(h) when we set the threshold $\Psi=200$, and the other parameters for different algorithms are listed in Table 4.





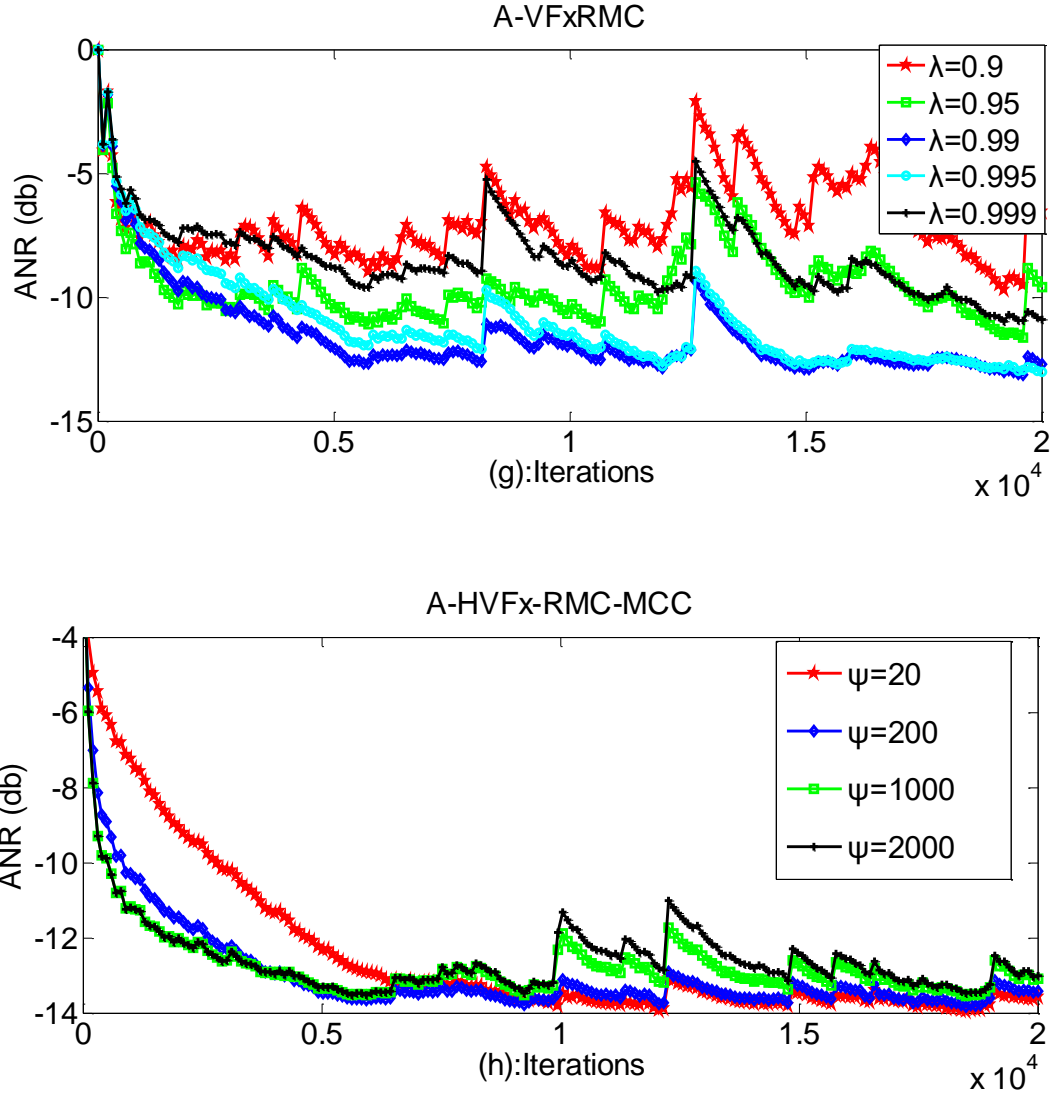


Fig.5 The ANR curves for (a) VFxLMS algorithm, (b) VFxlogCLMP algorithm, (c) VFxlogLMS algorithm, (d) RFsLMS algorithm, (e-f) A-VFxnMCC algorithm, (g) A-VFxFxRMC algorithm, (h) A-HVFxFx-RMC-MCC algorithm.

Fig. 6 shows the ANR learning curves of the three proposed MCC-based algorithms. As we can see, all of the developed algorithms are found to be stable for the impulsive noise with non-minimum phase secondary path in NANC. Both the A-VFxFxRMC and the hybrid HVFxFx-MCC-RMC algorithms converge faster than the A-VFxnMCC algorithm. With the similar convergence rate, the hybrid algorithm gives more robust result compared with the A-VFxFxRMC algorithm.

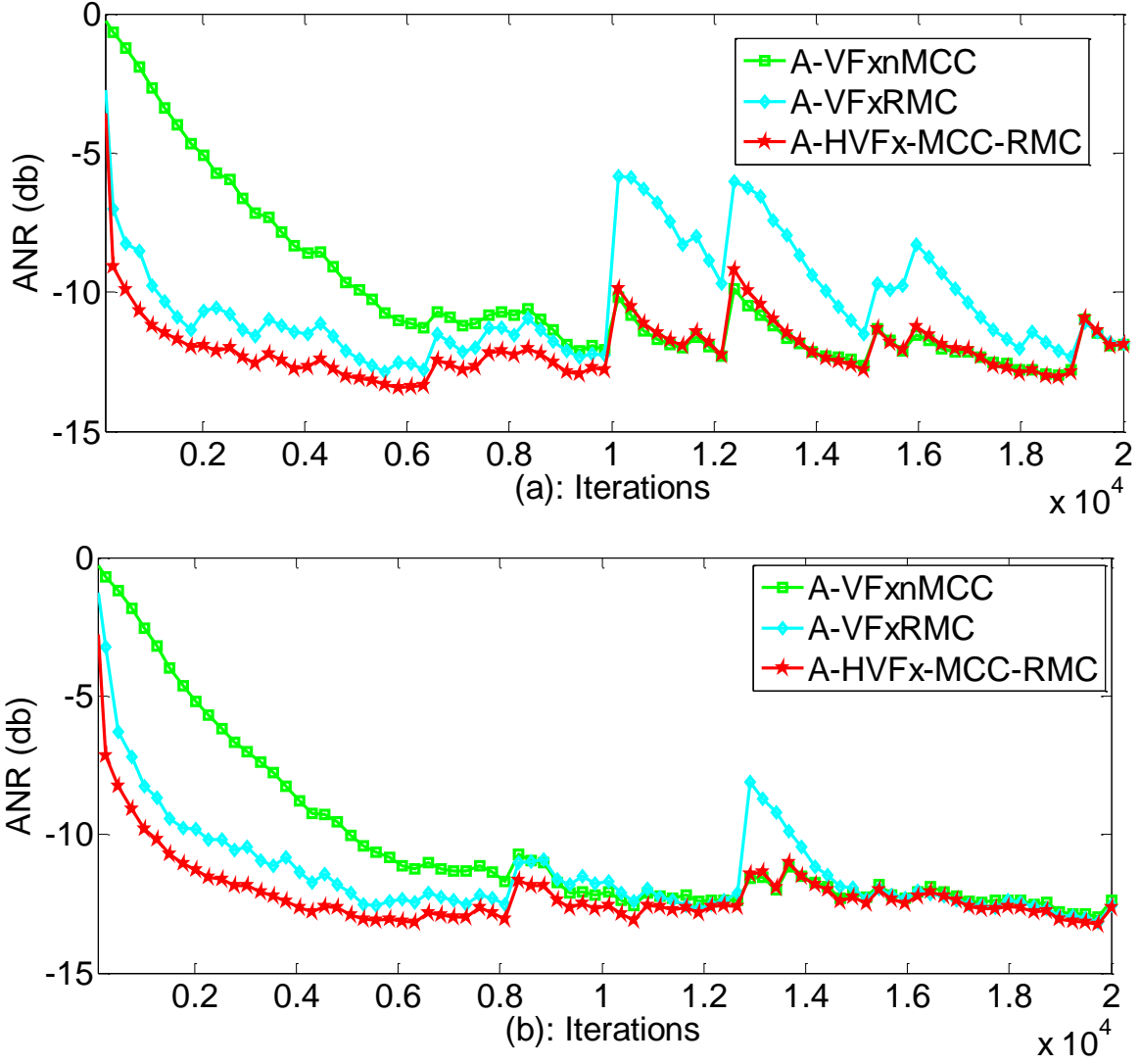


Fig.6 Performance comparison in an ANC system with minimum phase secondary path for impulsive noise condition. (a) $\alpha = 1.7$, (b) $\alpha = 1.8$.

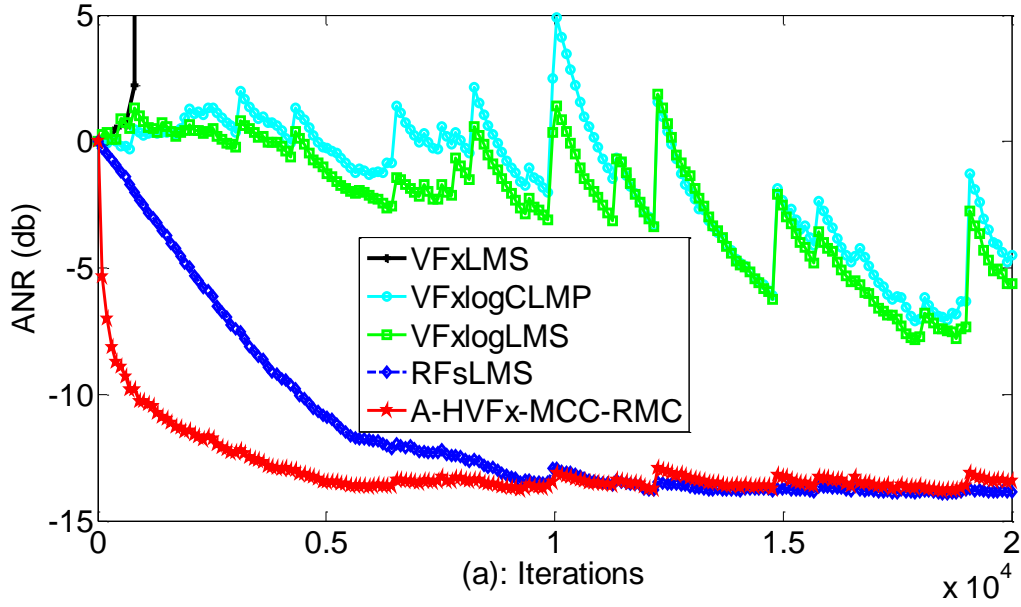
In addition, the computation time of the simulation relating to Fig. 6 is outlined in Table 5. It is seen that the elapsed time of A-VFxnMCC and hybrid HVFX-MCC-RMC algorithms is less than that of A-VxFRMC algorithms, confirming that the hybrid algorithm has the similar low computation burden with A-VFxnMCC. Furthermore, due to little difference of the ANR performance of all the developed algorithms, we will just use the hybrid HVFX-MCC-RMC algorithms to compare with the exiting algorithms in the following simulation experiments.

Fig. 7 depicts the convergence curves of adaptive algorithms in ANC domain for the noise shown in Fig. 4. It can be seen that the VFxLMS algorithm diverges in all cases where the impulsive noise appears. Unsurprisingly, the results obtained from other algorithms are

always stable. The proposed hybrid HVF_x-MCC-RMC and RFsLMS algorithms achieve a lower noise reduction than other algorithms with different noise. However, the proposed hybrid algorithm gives a fast convergence compared to RFsLMS algorithm.

Table 4

Initial parameters		
Algorithms	Step-size	Gaussian kernel
VFxLMS	$\mu = 1e - 8$	--
VFxlogCLMP	$\mu = 1e - 1$	--
VFxlogLMS	$\mu = 3e - 1$	--
RFsLMS	$\mu = 1e - 2$	$\sigma^2=16$
A-VFxnMCC	$\mu_1 = 1e - 2, \mu_2 = 1e - 3$	$\sigma^2=16$ (initial value)
A-VF _x RMC	$\lambda = 0.99$	$\sigma^2=16$ (initial value)
A-HVF _x -RMC-MCC	$\mu_2 = 1e - 3, \lambda = 0.99$	$\sigma^2=16$ (initial value)



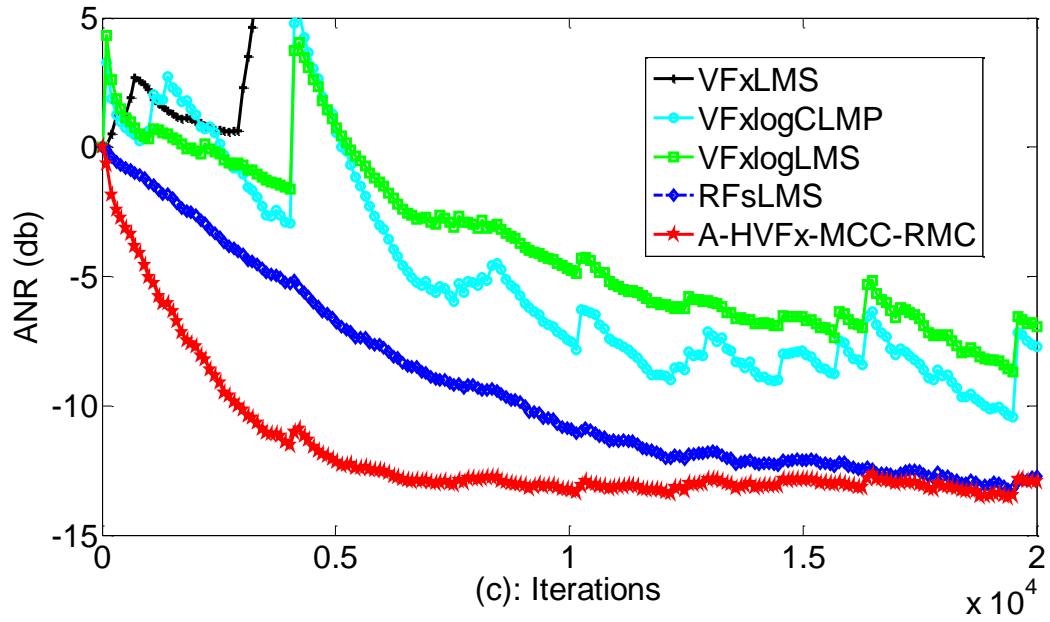
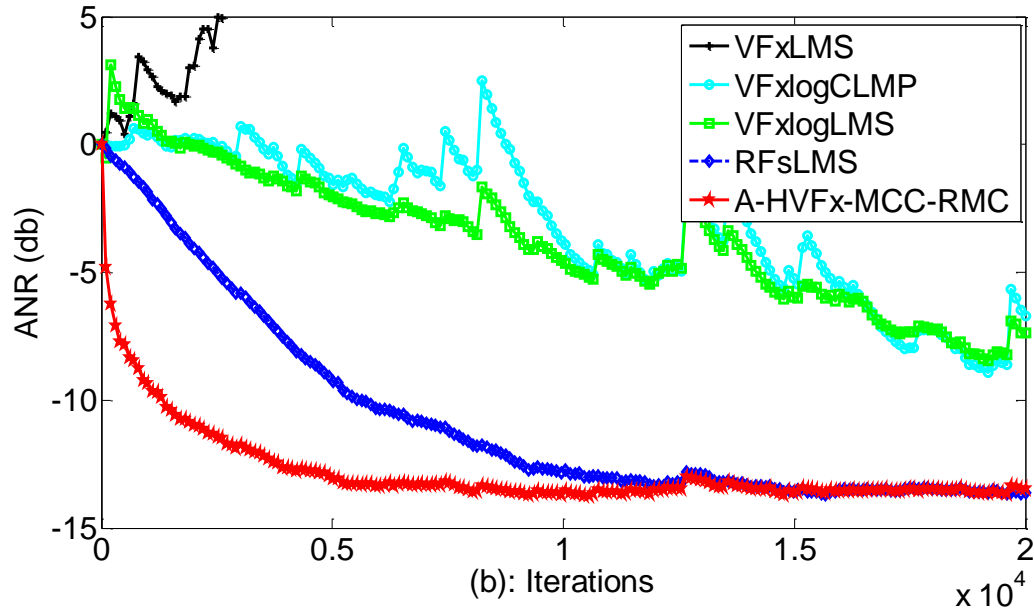


Fig.7 Performance comparison in an ANC system with existed algorithms in Experiment 1.

(a) $\alpha = 1.7$, (b) $\alpha = 1.8$, (c) $\alpha = 2.0$.

Table 5

The simulation computation time of experiment 1

Algorithms	Simulation computation time (s)	
	Case 1	Case 2
A-VFxnMCC	1.563	1.562
A-VFxRMC	3.853	3.535
A-HVFX-RMC-MCC	1.924	1.834

5.2 Experiment 2: Non-minimum phase secondary path and nonlinear primary noise

The previous section simulates the performance of multiple algorithms with the linear primary noise. In this experiment, we consider the primary path to be non-linear, and it can be given by

$$d(n) = v(n-2) + 0.08v^2(n-2) - 0.04v^3(n-1), \quad (43)$$

where $v(n) = x(n) * q(n)$, and $q(n)$ is the impulse response of the primary transfer function $Q(z) = z^{-3} - 0.3z^{-4} + 0.2z^{-5}$.

The other parameters such as the step-size and the primary transfer function in this experiment are the same as the previous simulation. The reference noise and primary noise applied in this experiment are shown Fig.8. One can see that the presence of a nonlinear primary path can severely increase the noise magnitude. As shown in Fig.9, the traditional VFxLMS algorithm still diverges. The proposed hybrid algorithm achieves a faster convergence and lower noise reduction, which can further verify the effectiveness of the proposed algorithms.

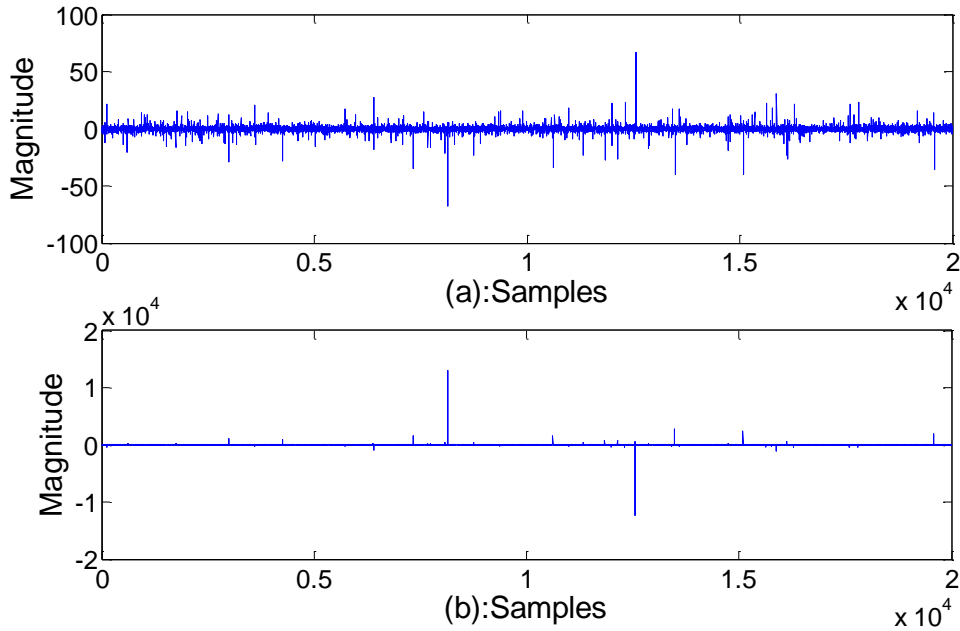


Fig.8. One of reference noise and primary noise in experiment 2. (a):reference noise, (b): primary noise.

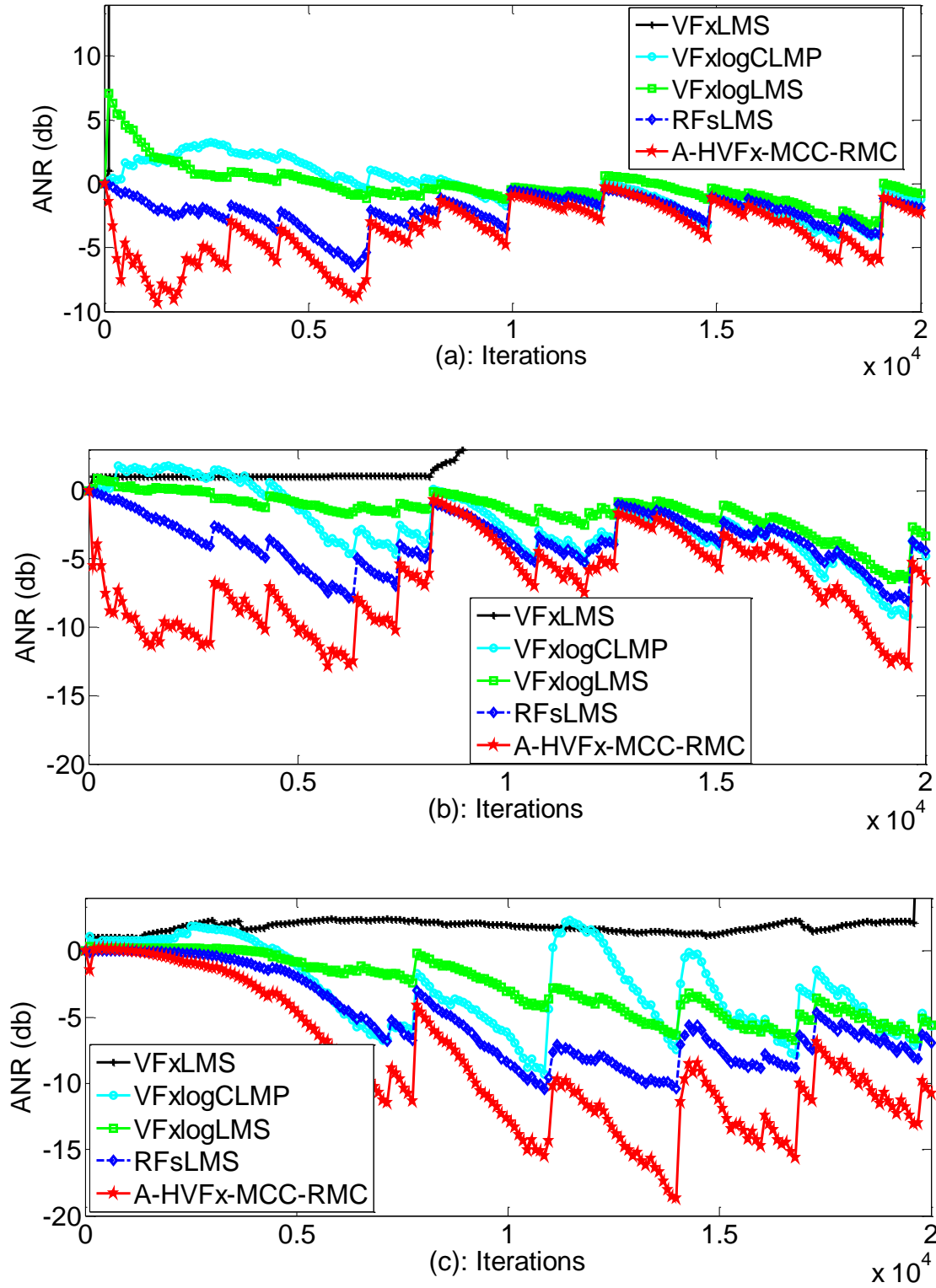


Fig.9 Performance comparison in an ANC system with existing algorithms in experiment 2. (a) $\alpha = 1.7$, (b) $\alpha = 1.8$, (c) $\alpha = 2.0$.

6. Conclusion

In this paper, the new Volterra filter NANC algorithms based on MCC are proposed to improve the stability and robustness in the presence of impulsive noise. Both the VFxnMCC and VFxRMC algorithms can effectively reduce impulsive noise. However, the VFxnMCC algorithm converges slowly while VFxRMC algorithm has a high computational complexity. The further proposed hybrid algorithm has a lower computational complexity than the VFxRMC algorithm, and gives a faster convergence than the VFxnMCC algorithm. In this work, we adaptively adjust the kernel size of MCC-based algorithms proposed. To evaluate the performance of the proposed algorithms, the results from the proposed algorithms are compared with VFxLMS, RFslms, VFxlogLMS and VFxlogCLMP algorithms, and it clearly demonstrates that the proposed algorithms have a faster convergence speed and lower noise reduction capability than the existing algorithms.

Acknowledge

The project is supported by the Science Fund of State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body No. 31615002. The authors wish to thank and Research Project of State Key Laboratory of Mechanical Systems and Vibration MSV201711 and Research Project of State Key Laboratory of Structural Analysis for Industrial Equipment (Grant No. GZ 1609) for the support.

References

- [1] Zhao H, Zeng X, He Z, Li T. Adaptive RSOV filter using the FELMS algorithm for nonlinear active noise control systems ☆. *Mechanical Systems & Signal Processing*. 2013;34:378-92.
- [2] Kuo SM, Morgan D. *Active Noise Control Systems: Algorithms and DSP Implementations*: John Wiley & Sons, Inc.; 1995.
- [3] Padhi T, Chandra M, Kar A. Performance evaluation of hybrid active noise control system with online secondary path modeling. *Applied Acoustics*. 2018;133:215-26.
- [4] Akhtar MT, Mitsuhashi W. Improving performance of FxLMS algorithm for active noise control of impulsive noise. *Journal of Sound & Vibration*. 2009;327:647-56.
- [5] Xiao Y, Ikuta A, Ma L, Khorasani K. Stochastic Analysis of the FXLMS-Based Narrowband Active Noise Control System. *IEEE Transactions on Audio Speech & Language Processing*. 2008;16:1000-14.

-
- [6] Sicuranza GL, Carini A. On the BIBO Stability Condition of Adaptive Recursive FLANN Filters With Application to Nonlinear Active Noise Control. *IEEE Transactions on Audio Speech & Language Processing*. 2011;20:234-45.
 - [7] Bambang RT. Adjoint EKF learning in recurrent neural networks for nonlinear active noise control. *Applied Soft Computing*. 2008;8:1498-504.
 - [8] Seng KP, Man Z, Wu HR. Nonlinear active noise control using Lyapunov theory and RBF network. 2000;2:916-25 vol.2.
 - [9] Das DP, Moreau DJ, Cazzolato BS. Adjoint nonlinear active noise control algorithm for virtual microphone. *Mechanical Systems & Signal Processing*. 2012;27:743-54.
 - [10] George NV, Panda G. A robust filtered-s LMS algorithm for nonlinear active noise control. *Applied Acoustics*. 2012;73:836-41.
 - [11] Tan L, Jiang J. Adaptive Volterra filters for active control of nonlinear noise processes. *IEEE Transactions on Signal Processing*. 2001;49:1667-76.
 - [12] Das DP, Panda G. Active mitigation of nonlinear noise Processes using a novel filtered-s LMS algorithm. *IEEE Transactions on Speech & Audio Processing*. 2004;12:313-22.
 - [13] Luo L, Bai Z, Zhu W, Sun J. Improved functional link artificial neural network filters for nonlinear active noise control. *Applied Acoustics*. 2018;135:111-23.
 - [14] Tan L, Jiang J. Adaptive second-order Volterra filtered-X RLS algorithms with sequential and partial updates for nonlinear active noise control. *Industrial Electronics and Applications, 2009 Iciea 2009 IEEE Conference on*2009. p. 1625-30.
 - [15] Zhou D, Debrunner V. Efficient Adaptive Nonlinear Filters for Nonlinear Active Noise Control. *IEEE Transactions on Circuits & Systems I Regular Papers*. 2007;54:669-81.
 - [16] Lu L, Zhao H. Adaptive Volterra filter with continuous l_p -norm using a logarithmic cost for nonlinear active noise control. *Journal of Sound & Vibration*. 2016;364:14-29.
 - [17] Liu W, Pokharel PP, Principe JC. Correntropy: Properties and Applications in Non-Gaussian Signal Processing. *IEEE Transactions on Signal Processing*. 2007;55:5286-98.
 - [18] Lu L, Zhao H. Active impulsive noise control using maximum correntropy with adaptive kernel size. *Mechanical Systems & Signal Processing*. 2017;87.
 - [19] Kurian NC, Patel K, George NV. Robust active noise control: An information theoretic learning approach. *Applied Acoustics*. 2016;117:180-4.
 - [20] Izanloo R, Fakoorian SA, Yazdi HS, Dan S. Kalman filtering based on the maximum correntropy criterion in the presence of non-Gaussian noise. *Information Science and Systems*2016. p. 500-5.
 - [21] Miyagi S, Sakai H. Performance comparison between the filtered-error LMS and the filtered-X LMS algorithms [ANC]. *IEEE International Symposium on Circuits and Systems*2001. p. 661-4 vol. 2.
 - [22] Haykin SS, Widrow B. *LEAST-MEAN-SQUARE ADAPTIVE FILTERS*2003.
 - [23] Yu F, Bouchard M. Recursive least-squares algorithms with good numerical stabilityfor multichannel active noise control. *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001 Proceedings*2001. p. 3221-4 vol.5.
 - [24] Wu L, Qiu X, Burnett IS, Guo Y. A recursive least square algorithm for active control of mixed noise. *Journal of Sound & Vibration*. 2015;339:1-10.
 - [25] Zeb A, Mirza A, Khan QU, Sheikh SA. Improving performance of FxRLS

algorithm for active noise control of impulsive noise. *Applied Acoustics*. 2017;116:364-74.

[26] Wertz W. Density estimation for statistics and data analysis - B. W. Silverman: Chapman and Hall; 1988.

[27] Wu Z, Shi J, Zhang X, Ma W, Chen B. Kernel recursive maximum correntropy. *Signal Processing*. 2015;117:11-6.

[28] Parzen E. On Estimation of a Probability Density Function and Mode. *Annals of Mathematical Statistics*. 1962;33:1065-76.

[29] Thai NL, Wu X, Na J, Guo Y, Tin NTT, Le PX. Adaptive variable step-size neural controller for nonlinear feedback active noise control systems ☆. *Applied Acoustics*. 2017;116:337-47.

[30] Slock DTM. On the convergence behavior of the LMS and the normalized LMS algorithms. *IEEE Transactions on Signal Processing*. 1993;41:2811-25.

[31] Sayed AH. Fundamentals of adaptive filtering. *Control Systems IEEE*. 2003;25:77 - 9.

[32] Leahy R, Zhou Z, Hsu YC. Adaptive filtering of stable processes for active attenuation of impulsive noise. *International Conference on Acoustics, Speech, and Signal Processing* 1995. p. 2983-6 vol.5.

[33] Wu L, He H, Qiu X. *An Active Impulsive Noise Control Algorithm With Logarithmic Transformation*: IEEE Press; 2011.